Journal of Manufacturing Systems 59 (2021) 345–354

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Making costly manufacturing smart with transfer learning under limited data: A case study on composites autoclave processing

Milad Ramezankhani [a], Bryn Crawford [a], Apurva Narayan [a,b], Heinz Voggenreiter [c], Rudolf Seethaler [a], Abbas S. Milani [a,*]

[a] Materials and Manufacturing Research Institute, University of British Columbia, Kelowna, V1V 1V7, Canada
[b] Department of Computer Science, University of British Columbia, Kelowna, V1V 1V7, Canada
[c] Institute of Structures and Design, German Aerospace Center (DLR), Stuttgart, Pfaffenwaldring 38-40, D-70569, Germany

ARTICLE INFO

Keywords:
Intelligent manufacturing
Transfer learning
Limited data
Autoclave processing
Aerospace composites

ABSTRACT

The integration of advanced manufacturing processes with ground-breaking Artificial Intelligence methods continue to provide unprecedented opportunities towards modern cyber-physical manufacturing processes, known as smart manufacturing or Industry 4.0. However, the "smartness" level of such approaches closely depends on the degree to which the implemented predictive models can handle uncertainties and production data shifts in the factory over time. In the case of change in a manufacturing process configuration with no sufficient new data, conventional Machine Learning (ML) models often tend to perform poorly. In this article, a transfer learning (TL) framework is proposed to tackle the aforementioned issue in modeling smart manufacturing. Namely, the proposed TL framework is able to adapt to probable shifts in the production process design and deliver accurate predictions without the need to re-train the model. Armed with sequential unfreezing and early stopping methods, the model demonstrated the ability to avoid catastrophic forgetting in the presence of severely limited data. Through the exemplified industry-focused case study on autoclave composite processing, the model yielded a drastic (88%) improvement in the generalization accuracy compared to the conventional learning, while reducing the computational and temporal cost by 56%.

1. Introduction

Recent advances in new technologies such as Cyber-Physical Systems (CPS), Internet of Things (IoT), Artificial Intelligence (AI) and Machine Learning (ML) have enabled the emergence of the modern era manufacturing, known as smart manufacturing. Particularly, recent achievements of novel ML algorithms such as deep learning [1] have enabled the manufacturing industry to develop adaptable, customizable and responsive manufacturing platforms by integrating the highly advanced manufacturing infrastructure with accurate and reliable predictive models [2–4]. This can yield economic growth and superiority for advanced manufacturing industries in the modern global market, where agile design and production of new products — in response to customer's needs — play a central role [5].

Despite the much-touted capabilities of machine learning (ML) methods, when it comes to implementation of complex data-driven predictive platforms in expensive manufacturing settings, often there exist major restrictions (e.g., data availability, data consistency, model complexity and model architecture, etc.), which if not addressed accordingly, can limit the models' performance, and consequently degrade the level of reliance and trust amongst the users. In the context of smart manufacturing, the key conditions underlining the aforementioned limitations of conventional ML methods may be summarized into two categories: the lack of sufficient data, and probable shifts and deviations in the production line [6]. The survival of smart manufacturing enterprises hugely relies on the extent to which they can serve as configurable and adaptable facilities against the inherent changes and shifts of markets [6].

Enabling the conventional manufacturing processes to produce on-demand personalized products in response to the dynamic market needs is one of the key characteristics of smart manufacturing [7]. According to the National Institute of Standards and Technology (NIST), smart manufacturing is defined as "fully-integrated, collaborative manufacturing systems that respond in real-time to meet changing demands and conditions in the factory, in the supply network, and customer needs." [2]. However, conventional ML models cannot offer

* Corresponding author.
E-mail address: abbas.milani@ubc.ca (A.S. Milani).

https://doi.org/10.1016/j.jmsy.2021.02.015
Received 30 July 2020; Received in revised form 24 February 2021; Accepted 25 February 2021
Available online 19 March 2021

the high level of adaptability required for smart manufacturing. The underpinning assumption for such models requires the training (e.g., the collected historical data from previous processes) and the test (e.g., unseen data from a new process) data, to be drawn from a similar probability distribution and that training data be comprehensive and descriptive of the whole input space. This premise makes conventional machine learning model quite vulnerable to even minute changes in the manufacturing process, resulting in erroneous predictions [8,9]. To tackle this issue, one unrealistic solution would be to generate large training datasets out of every new production setup and then train a new ML model from scratch. The problem with this approach is, however, that collecting new data points in some manufacturing applications is expensive and time-consuming. Even collecting enough data points for one new process does not universally resolve the issue, as future modifications in the process again seek a daunting task of data generation.

1.1. Overview of emerging ML applications in composites manufacturing

Focusing on composites materials and processing, ML models have been mainly implemented to develop optimization design spaces for the rapid identification of optimum mechanical properties. Arian Nik et al. [10] used finite element (FE) data to construct ML models for fast design optimization. In an attempt to alleviate the dependency on time-consuming FE simulations, Zimmerling et al. [11] utilized forming simulation data to construct an ML model capable of estimating the in-plane shear angle. Within a global sensitivity framework, Balokas et al. [12] developed a neural network model that estimates the elastic properties of braided composites in automated manufacturing processes. Recently, deep learning models have received increasing attention [13]. In [14], Khan et al. used low-frequency structural vibration responses of composite laminates as the input of a convolutional neural network (CNN) for predicting in-plane and through-the-thickness delamination. In the studies [15,16], in order to establish a structure-property linkage, CNN models were developed to predict the mechanical properties using composites' microstructure images. Again, the aforementioned ML frameworks have been developed based on the restricting assumption that a sufficient amount of data is available for the training process of the model and data distributions remain unchanged (e.g., no modifications/shifts in the manufacturing process, images format, etc.). Such underlying assumptions are often violated in real-world manufacturing scenarios, resulting in a significant decline in the performance and generalizability of the developed ML models. This in turn impedes the successful implementation of advanced predictive algorithms in smart manufacturing applications as they are prone to continual changes in process conditions and production setups [2].

1.2. Objective and organization of this work

In order to combat the above-referenced shortfalls of the conventional ML applications in smart manufacturing, a robust learning system is deemed necessary with the following specifications: (i) it should be capable of exploiting transferrable (useful) knowledge from auxiliary sources of data and implement it in the current (new) training process, under highly limited data availability; and (ii) be immune against inherent process shifts in the given manufacturing platform. In this article, a transfer learning (TL) framework is aimed to attain the above specifications and address the manufacturers' need for an agile and robust predictive modeling tool for fast design space exploration and decision-making tasks. The proposed TL model, theoretically, does not require the training and test data to be drawn from the same process distribution, nor does it need large datasets for achieving high accuracy [17]. It leverages the similarity between the existing historical data (source) and the limited observations collected from the new process settings (target). It updates the target model with a low-level representation of the source model, which in turn facilitates the fine-tuning (adapting) of the target model with available data [18,19]. The applicability of the approach has been illustrated via a case study in the autoclave manufacturing of aerospace composites.

The rest of the manuscript is organized as follows. Section 2 reviews the autoclave curing process in composites manufacturing and discusses the current limitations in predicting the thermal history of the parts under limited data. Section 3 presents the definition and overall methodology of TL and overviews its crucial role in developing reliable predictive models for smart manufacturing. Section 4 describes the proposed modeling framework; Sequential unfreezing, an accuracy boosting method embedded to avoid overfitting and catastrophic forgetting, is also discussed. Numerical results on the validity of the method are presented in Section 5, and Section 6 outlines the summary, limitation and future direction of the work.

1.2.1. Expected application novelty

In the realm of Industry 4.0, which is still a vision in advanced aerospace manufacturing, shifts in a given costly process can occur e.g. by assigning a new cure cycle design for autoclave processing of large composites structures [20]. Such complex and sensitive manufacturing processes, however, are severely vulnerable to design shifts, in a way that even a minute deviation from the process optimal setting can yield catastrophic consequences (e.g., unsatisfactory product properties or defect) and sometimes, irretrievable losses [21]. This level of sensitivity requires careful predictive modeling pipelines embedded in the cyber-infrastructure of the smart factory settings. The present case study shows how to remedy the inadequacy of conventional ML methods in composite process design and manufacturing contexts by developing a *cross-process* TL framework. This approach leverages the knowledge from the previous cure process, uses it for constructing a customized predictive model with a small dataset, and obtains satisfactory precision when applied to the new curing process. The trained predictive model can be used to identify the optimal process specifications to be implemented in the producibility trials during trade-off studies and decision making towards detailed designs. To the best of the authors' knowledge, this field of research in composite field has remained rarely explored.

2. Case study: composites autoclave manufacturing

Superior mechanical properties, such as high specific strength and stiffness, have made fiber reinforced polymer composites the dominant choice in a myriad of advanced manufacturing applications, especially where lightweight structures are critical (e.g., in aerospace and automobile industries) [22]. However, manufacturing composite structures is a complex process with high levels of uncertainty. In fact, the resulting properties of the manufactured part heavily depend on both the raw material characteristics and process conditions, in a way that the existence of a small deviation from the optimal process setting might result in defects or inadequate performance characteristics [23]. Particularly in modern aerospace manufacturing applications, thermoset prepregs are used as the raw material for fabricating the majority of the composite structures. A thermoset prepreg is prepared by pre-impregnating thin layers of fibers (e.g., carbon or glass) with e.g. epoxy, the most common type of high-performance resin employed in aerospace applications. At the time of manufacturing the part, the thermoset prepreg sheets are taken out of the freezer (to avoid unintentional developments in the material's degree of cure, the prepreg needs to be kept in sub-ambient temperatures [24]), cut into desired shapes, and laid-up on the tool with the required ply orientations. Then, the laid-up part is placed in a vacuum bag to eliminate trapped air and consolidate the layers of prepreg, all of which prepare the part to be cured in the autoclave [23].

During the curing process in an autoclave vessel, specified pressure and temperature cycles (i.e., inert nitrogen gas heated by electrical elements) are applied to the part over a specific amount of time, which in turn converts the uncured stacked layers of thermoset prepreg to a consolidated and cured composite part [24]. Among all the manufacturing steps of the production workflow, the process of curing

the prepregs in the autoclave plays the most significant role in developing many crucial outcomes, such as the mechanical properties and geometry (Fig. 1).

Thermal history components of an aerospace composite part during the curing process, namely, thermal lag and exotherm, have been proven to be key metrics for achieving the part's required physical (e.g., reducing the occurrence of cure-induced voids) and thermal (e.g., reducing thermal undershoot that causes an inadequate degree of cure) specifications. In turn, maintaining the part's thermal history within the allowable envelope (e.g., the manufacturer's recommended cure cycle - MRCC) can yield the desired mechanical, chemical and physical properties (e.g., high modulus) [25,26]. The exotherm is defined as the maximum temperature of the part during the curing process. The polymerization of the resin is an exothermic process and can generate a substantial amount of heat. This can increase the temperature of the part, which if it exceeds the allowable exotherm limit, can damage the part [23]. The thermal lag, on the other hand, specifies the deviation of the part's temperature from the cure cycle's temperature settings and needs to be limited during the curing process. High values of thermal lag demonstrate divergence from the defined cure cycle, resulting in an under- cured part [23].

The thermal history of the part itself is governed by the autoclave process parameters. For instance, increasing the velocity of the gas flow in the autoclave yields a higher heat transfer coefficient (HTC) between the part and autoclave [24]. Higher values of HTCs help the part to dissipate the generated heat more quickly during the resin's polymerization process, and hence, prevent the undesirable increase in the part maximum temperature (exotherm). The mechanical and physical properties of the tooling material considerably influence the thermal history of the part. On one hand, the thermal properties of the tool determine the conductive heat flow into and out of the part during the curing process. On the other hand, the geometry of the tool modifies the gas flow profile, resulting in different HTCs along the part-tool system. This justifies in practice the necessity of studying/modeling the relationships between the process conditions and the thermal properties of the part to obtain the optimal process parameters which result in parts with desirable properties.

Previous works have attempted to capture the process underlying complex material behavior and predict the final properties of the finished part, as well as defects using conventional machine learning models. It was assumed that the available limited data is enough for model training purposes and the data distributions remain unchanged over time [27,28]. Collecting sufficient data points for training a predictive model on such a complex process, however, is quite challenging in practice. The capital and operational costs for the curing process are high and it can take between 3–10 h to complete [23]. Moreover, in a smart composites manufacturing setting (composites 4.0), when a new cure cycle is introduced to the system, the in-use prediction model does not perform well against the new unseen data, especially if the old and new cure cycles are dissimilar (see more details in Section 5). In fact, the properties and thermal profile of the cured part are highly associated with the corresponding process parameters, and thus, it requires a high level of adaptability in the ML algorithm. As will be discussed in more detail in the next sections, with the use of advanced machine learning methods, the underlying mapping between the curing process parameters — e.g. the HTC of both part and tool sides, tooling properties, part and tool thicknesses, and autoclave working fluid heating rate — and the process thermal management outcome (namely, thermal lag and exotherm [23]) under a new cure cycle setting with limited data, can be investigated.

3. Transfer learning

Generally speaking, the performance of ML algorithms vastly relies on a general assumption about the data: the training and test data are drawn from the same probability distribution in a way that the input feature space of two datasets along with their data distribution characteristics be exactly alike. Such a restrictive premise does not hold in many real-life applications [8] and therefore, due to the lack of comprehensive and balanced data, conventional ML models cannot always afford the high level of adaptability required in smart manufacturing [29]. Besides, in many tasks of interest (e.g., the autoclave curing process of composite materials), the amount of on-hand data is painfully limited. This makes the task of training more difficult, particularly in the case of implementing "data-greedy" approaches such as deep learning methods, which often require excessive labeled data [30]. In order to make machine learning approaches more applicable to advanced manufacturing problems, a new learning framework to address the current learning models' limitations needs to be developed. This is where *transfer learning* comes into the picture. In avoiding the need for expensive data generation efforts by transferring useful knowledge from other related domains, TL provides a robust learning

Fig. 1. Interior of an industrial autoclave example in composite manufacturing (photo courtesy of Composites Research Network).

framework in the case of discrepancies between training and test datasets [6]. In the context of smart composites manufacturing, when a shift happens in the process configuration (e.g., introducing a new cure cycle for producing a new composite part), the underlying mapping between the input and output variables will change inevitably, making conventional machine learning models inefficient for prediction tasks on the new data distribution. The TL models, however, are able to properly adapt to the new data structure and maintain their high performance [31,32]. As example applications of TL, Wen et al. [29] proposed a deep TL model that used auto-encoders and maximum mean discrepancy to exploit similar low-level feature representations of source and target data and applied it in production progress prediction in IoT-Enabled manufacturing. Xu et al. [33] implemented TL to transfer the learned knowledge from a virtual fault diagnosis digital twin simulation to a physical maintenance system. Sun et al. [34] used deep TL methods to predict the cutting tools' useful life span with the transfer of knowledge from historical failure data. Ferguson et al. [35] presented a TL-based CNN to improve the task of manufacturing defects detection by learning features from two large image datasets. Li et al. [36], through a multi-class adversarial domain adaptation procedure, developed a semi-supervised transfer learning framework (i.e., no *labeled* target data) in order to achieve a class-level alignment between the source and target domains in machinery fault diagnosis.

Mathematically, in a TL framework, a task of interest (target) is learned by incorporating the knowledge from an already learned task (source) [37]. Each machine learning problem comprises a domain and a task [38]. A domain, $\mathscr{D}$, is defined by two components, an input feature space, $\mathscr{X}$, and a marginal probability distribution, $P(X)$, where $X = \{x_1, ..., x_n\} \in \mathscr{X}$. For a given domain, $\mathscr{D} = \{\mathscr{X}, P(X)\}$, a task is consists of two components, a label space $\mathscr{Y}$ and an objective predictive function, $f(\cdot)$, which is meant to be learned by the learning method using the training data, $\{x_i, y_i\}$, where $x_i \in X$ and $y_i \in \mathscr{Y}$. Once the $f(\cdot)$ is learned, it can be implemented to predict the associated label/prediction value $f(\hat{x})$ of an unseen data, $\hat{x}$. TL can be defined as follows: Given a source domain dataset, $D_s = \{(x_{S_1}, y_{S_1}), ..., (x_{S_{n_s}}, y_{S_{n_s}})\}$, and a learning task, $\mathscr{T}_S = \{\mathscr{Y}_S, f_S(\cdot)\}$, TL is the process of improving the target predictive function $f_T(\cdot)$ in a target domain dataset, $D_T = \{(x_{T_1}, y_{T_1}), ..., (x_{T_{n_T}}, y_{T_{n_T}})\}$ by levering the knowledge in $\mathscr{D}_s$ and $\mathscr{T}_S$, where at least one of the following conditions applies: $\mathscr{D}_s \neq \mathscr{D}_T$ or $\mathscr{T}_S \neq \mathscr{T}_T$ [37]. The former inequality requires that either $X_s \neq X_T$ or $P_S(X) \neq P_T(X)$ in the source and target domains setting. Similarly, the latter condition implies that either $\mathscr{Y}_S \neq \mathscr{Y}_T$ or $f_S(\cdot) \neq f_T(\cdot)$. The special case where domains and tasks of the source and target are the same transforms the problem to a conventional machine learning approach. Depending on how the source and target learning structures differ, the availability of *labeled* target data, and the type of the knowledge being transferred, various approaches and methods have been introduced to accomplish the learning task via knowledge transfer [37,39].

4. Proposed framework and implementation

In the case of composites autoclave curing process, the abundant historical data (source) from a conventional classic cure cycle (e.g., standard one-hold) can be used to enhance the accuracy of the model developed by the limited data (target) generated from a non-classic/complex cure cycle (e.g., a new two-hold cycle). This can significantly decrease the dependency of the advanced manufacturing models to excessive data and therefore, robust predictive tools can be developed using limited data. The proposed general workflow for such TL for manufacturing is illustrated in Fig. 2, and discussed in detail in the following sub-sections.

4.1. Source and target datasets generation

The data for this case study was generated for two cure cycles in the process of manufacturing AS4/8552 aerospace-grade composite material, using RAVEN process simulation package [40] (the first cure cycle is assumed to be the source with abundant data availability, and the second one – the target – with limited data availability). RAVEN performs thermal, physical and mechanical property estimation for given part configurations and cure cycles, and outputs the thermal history of the composite material. Further, this model can be run parametrically, facilitating the exploration of large design spaces. For the one-hold cure cycle (CC1), the curing process starts with increasing the temperature from 20 °C to 180 °C (with a specified constant heating rate) and continues with an isothermal holding step at 180 °C for 2 h. The process is

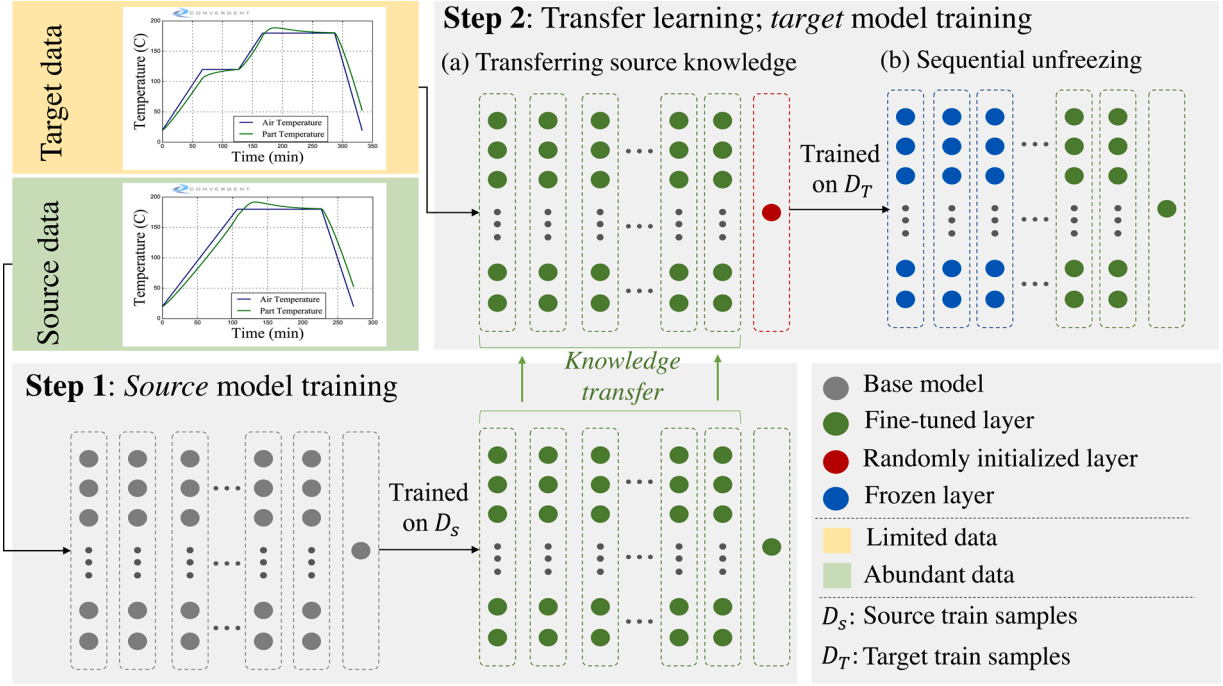

Fig. 2. The framework of the cross-process TL method in smart manufacturing.

completed by cooling down the material to room temperature with a 3.5 °C/min constant heating rate. Fig. 3.a demonstrates the CC1 specifications. The two-hold cure cycle (CC2) on the other hand, as shown in Fig. 3.b has two hold steps. It begins with a temperature ramp to 120 °C, continues with a temperature hold for 1 h, followed by a second temperature ramp to 180 °C. The temperature again is held at 180 °C for 2 h and is then decreased to 20 °C in a cool-down process with a 3.5 °C/min heating rate.

The input feature space for both source and target datasets consists of two types of variables, namely design specification variables including the tool material, tool thickness, part thickness, and the manufacturing process parameters which are heat rate, top- and bottom-side HTCs, as summarized in Table 1. The thermal lag and exotherm are selected as the outputs of interest for this study. Fig. 4 represents a schematic of a part being cured in an autoclave along with the affecting design and process variables. Among the cure cycles' specifications, namely heating rate and hold time, the heating rate of cycles' ramp is considered as a design process variable (input) as it noticeably controls the thermo-chemical response of the cured part. Thermal lag, for instance, is the product of the cure cycle ramps and can be strongly influenced by adjusting the heating rate [23]. On the other hand, a suboptimal heating rate can excessively increase the exotherm response [41]. In other words, given a specific part thickness, an increase in heating rate can lead to higher exotherm [20]. Hold times, however, are specified according to the manufacturer's recommendation and kept unchanged during the data generation process.

Table 1
Curing process input variables and the corresponding values.

Input variables	Values	
Tool material	6061 Aluminum; AS4/8552 Composite; Invar 36; SAE 1020 Steel	
	Min	**Max**
Took thickness (mm)	2.5	20
Part thickness (mm)	2.5	20
Heat rate (°C/min)	1	5
Top-side HTC (W/m^{2}K)	10	125
Bottom-side HTC (W/m^{2}K)	10	125

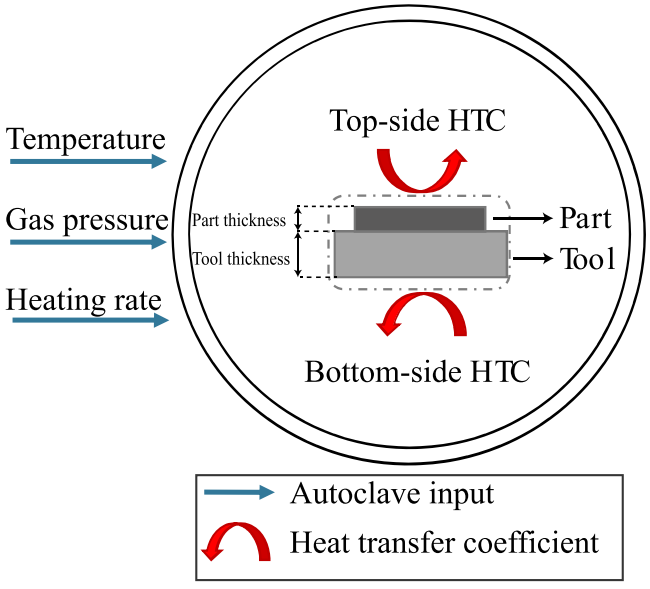

Fig. 4. Schematic of autoclave curing process in composites manufacturing.

4.2. Feature-based transfer learning

The learning framework in Fig. 2 is inspired by the idea of the transferability of learned features from one neural network to another. Yosinski et al. [43] showed that when trained on natural images, the first layers of neural networks learn similar low-level features and this is general to all networks regardless of their cost function and input dataset. In TL, the generality of features within networks can play a central role, in a way that the learned features can be transferred to the target network to improve the training process on the target dataset. Additionally, it has been shown that the transferred knowledge contains not only the learned feature representations but also the *low-level* data statistics. The latter (e.g., information about data distribution), which is generic to any type of data (e.g., structured, image, text, etc.), conveys crucial information for training a predictive model, and if it is transferred to the target network, it can speed up the model convergence and alleviate the large data requirements [42,43].

Since the feature space being investigated in this paper is structured (i.e., tabular data) and less complex than image data, a Fully Connected Neural Network (FCNN) with a few hidden layers is implemented (see Section 5.1 for the network specifications). Herein, initially, a base (source) network f_S (i.e., an FCNN) is trained using sufficient data from the source dataset D_s (Fig. 2, step 1) by minimizing the regression loss function

$$\mathscr{L}_R(f_s) = E[\ell(f_s(X_s),\ Y_s)], \tag{1}$$

where $E[\cdot]$ represents the statistical expectation and ℓ is a suitable loss function (see Section 5.1. for the details). Next, all the layers' weights except that of the last layer (regressor) are transferred to the new network (target) with the same architecture (Fig. 2, step 2.a). In a TL framework, when the target model is initialized with the fine-tuned

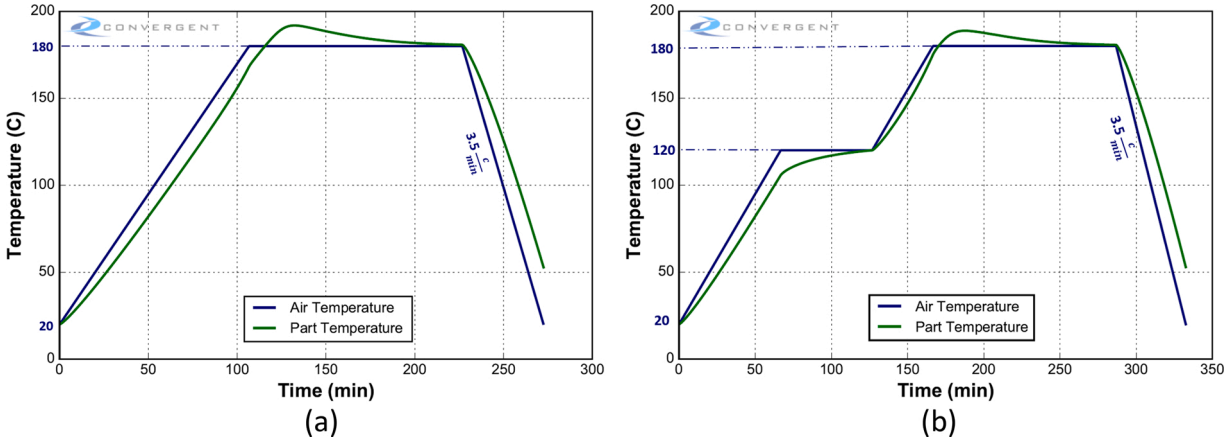

Fig. 3. Chosen autoclave cure cycles in the composites manufacturing curing process. (a) one-hold cure cycle (CC1); (b) two-hold cure cycle (CC2).

weights of the source network, the target model essentially remains in the "same basin in the loss landscape"; i.e., in the weights' optimization space [42]. This way, the maximum performance of the target model can be achieved by only taking a few gradient steps (i.e., fine-tuning) using a small amount of data toward the optimal weights of the new related task [44]. The last layer of the target network (the one specific to the new manufacturing process) is initialized randomly. The objective is to learn a predictive network f_T (consists of a feature representation mapping and a regressor) that accurately estimates the target (CC2) outputs, namely thermal lag and exotherm. Since the conventional learning approaches are not practical due to the limited target data, the framework instead transfers the learned feature representation of the source model to the target model and then adapts it accordingly via fine-tuning the weights using limited target data D_T

$$f_T = \text{fine-tuning}(f_S|D_T) \tag{2}$$

The resulting model then can be used for target data prediction tasks.

4.3. Sequential unfreezing

Once the target network is constructed, layers are sequentially fine-tuned (Fig. 2, step 2.b). During the training of the target network, the transferred layers can be either fine-tuned to adapt to the new task by error backpropagation or kept unchanged (frozen). The latter requires the remaining unfrozen layers to contribute to the target task training and turn into the task-specific layers of the target network. The complexity of the network and the size of the target dataset play a decisive role in whether to fine-tune or freeze the transferred layers. In the case of an extremely small dataset (e.g. in advanced composite structures manufacturing), or when a dense network with excessive parameters is selected to capture high-dimension and significantly non-linear system behavior, freezing the layers might be a good choice, as otherwise overfitting can occur and the generalization performance of the model may decline.

Since the randomly initialized layers generate large error gradient during the initial training iterations, it destructively updates the finely-tuned weights of the previously-trained transferred layers, with these high-magnitude changes during backpropagation. This phenomenon, known as catastrophic forgetting [45], causes the target network to "forget" the learned knowledge from the source and consequently, does not yield satisfying performance. To tackle this issue, the sequential unfreezing approach is proposed [46,47]. First, the randomly initialized layer is fine-tuned for n epochs while the weights of the other layers are frozen. At epoch $n+1$, all or some of the transferred layers become unfrozen (the optimal setting can be achieved by monitoring the model performance; Section 5.2) and the model is fine-tuned (as shown in Eq. (2)) for m epochs until convergence is achieved. During epochs $n+1$ to $n+m$, a small learning rate is chosen for the optimizer in order for preserving the transferred knowledge as much as possible. However, to more effectively explore the parameters space, a bigger learning rate is employed initially for fine-tuning the randomly initialized layers during the first n epochs.

5. Results and discussion

In this section, the predictive performance of the modeling framework in Section 4 is measured and compared with a conventional ML method that does not leverage TL. Furthermore, the effect of domain shifts in the performance of the already developed predictive model is investigated, via a model trained on CC1 data, but tested on a CC2 dataset. Finally, the effect of incorporating the sequential unfreezing method and early stopping in the TL model is investigated.

5.1. Cross-process transfer learning

In the initial analysis, the performance of the conventional ML and the TL models were compared for the current case study. 44,000 data points for each cure cycle were collected using RAVEN software (by varying the process parameters as outlined in Table 1). Once a Z-score normalization (pre-processing) is applied, the data points were randomly divided into a training set (75%, of which, 25% is used for validation – 18.75% of the total dataset) and a test set (25%). The model used the training dataset to update the weight of each neuron with the goal of minimizing the loss function, which quantifies the error between the predictions and observations. Then the validation data was used to measure the performance of the model on the validation set and monitor any potential sign of the overfitting. Finally, once the training and validation are complete, the test data (unseen to the model) was devised to assess the generalization error of the model. To mimic a real-life scenario in advanced composite manufacturing, it was assumed that only 500 training data is available from CC2 (target data) and the remaining was kept for generalization assessment of the models. For computational analyses, a PC with Intel Core i7-8700 CPU 3.6 GHz, and 32 G RAM was utilized. First, a neural network on CC1 full dataset (75% for training/validation and 25% for testing) was developed to construct a model on CC1 to be used as the source model for in the TL framework (CC1-Upper bound). In addition, in order to examine the performance of a predictive model in case of a shift in the manufacturing process, the CC1-Upper bound model is tested against the test data from the CC2 dataset. Next, using the proposed TL framework, the layers from the source model were transferred to the target model, which then was fine-tuned using the 500 training data points from CC2. In order to evaluate the performance of the TL method, a neural network was trained without TL, i.e., solely on the same 500 training data from CC2 for comparison (CC2-Limited). Finally, to estimate an upper-bound (imaginary ideal) performance in the target task, a model was trained on the full target dataset (44,000 data points) from CC2 (CC2-Upper bound). For all the developed models, feed-forward neural network architecture with 6 fully-connected hidden layers and 10 neurons per layer was implemented. Due to its self-normalizing properties and in order to prevent the vanishing/exploding gradients problem, the Scaled Exponential Linear Unit (SELU) activation function was used (Eq. (1)), where α and λ are pre-determined parameters ($\alpha \approx 1.673$ and $\lambda \approx 1.050$) and are solved to yield a self-normalizing neural network (i.e., during training, the layers' output maintains zero mean and unit variance) as described in the original paper [48]:

$$selu(x) = \lambda \begin{cases} x & \text{if } x > 0 \\ \alpha e^x - \alpha & \text{if } x \leq 0 \end{cases} \tag{3}$$

The output layer contains one neuron outputting the model's outcome on the process thermal history (either thermal lag or exotherm). The Mean Squared Error (MSE) was used as the loss function which is defined as:

$$J = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4}$$

where n is the number of training data, y_i denotes the observed data, $\hat{y}_i$ is the model's prediction, and the subscript i represents the i^{th} observation.

Table 2 summarizes the results of the models' analyses. The Root Mean Square Error (RMSE) and R-squared (R^2) are utilized as the models' evaluation metrics:

$$RMSE = \sqrt{\frac{1}{k}\sum_{i=l}^{k}\left(y_i - \hat{y}_i\right)^2} \tag{5}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{k}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{k}(y_i - \bar{y})^2} \tag{6}$$

where k and $\bar{y}$ represent the size and average of the test dataset, respectively.

Table 2
Comparison of conventional ML and TL approaches on predicting CC2 test data. RMSE (MSE) values are reported for the generalization loss.

Modeling cases	Generalization Loss (°C)		R-Squared		Training Time (s)	
	Exotherm	Lag	Exotherm	Lag	Exotherm	Lag
CC2-Upper bound[1]	0.162 (0.026)	0.21 (0.0441)	0.999	0.999	863	1318
CC2-Limited[2]	17.273 (298.35)	14.493 (210.05)	0.91	0.876	143	285
Transfer learning[3]	**1.91 (3.648)**	**0.384 (0.147)**	**0.996**	**0.998**	**53**	**98**
CC1-Upper bound[4]	155.99 (24,332)	90.174 (8131.4)	0.374	0.711	774	1003

[1] When the entire (hypothetically available) data of CC2 is used to train a conventional ML and predict the test (future) data of CC2.
[2] When the limited available data of CC2 is used to train a conventional ML and predict the test (future) data of CC2.
[3] When the entire available data of CC1 is combined with limited available data of CC2 to train a TL model and predict the test (future) data of CC2.
[4] When the entire available data of CC1 is used to train a conventional ML and predict the test (future) data of CC2.

The evident poor performance of CC1- Upper bound model against CC2 data demonstrates how vulnerable prediction models are to changes in cure cycle specifications. In other words, it highlights the necessity of using TL for dealing with such shifts in manufacturing applications. Compared to the conventional ML model (CC2-Limited), the proposed TL framework achieved a significantly higher performance (88% decrease in RMSE of exotherm prediction and 97% decrease for lag). While the generalization loss and the R^2 of the proposed model reached a satisfactory level in comparison with the upper-bound model (CC2-Upper bound), the training time was decreased drastically too, by 62% for the exotherm outcome (56% for lag). Fig. 5 shows the Pareto plot of CC2-upper bound (i.e. idealistic) and limited data models via conventional ML, as well as the targeted cross-process TL models along with the corresponding error frequencies. In the case of the TL models, both the count and magnitude of the erroneous predictions have decreased when compared to the conventional learning models, in the presence of limited data (i.e. less deviation from the 45-degree reference line). It can also be observed that the lag model performs relatively better than the exotherm model. Recalling the partial differential equations that govern the composites curing process could shed more light on the reason for such behavior. As shown in [49], in comparison to exotherm, there are fewer related system parameters involved in determining the value of thermal lag. For instance, the calculation of exotherm requires the analysis the *transient phase* of the curing process for which often no good approximation is possible [49]. Additionally, the prediction of the exotherm requires determining the internal exothermic heat generation of the part during curing ($\dot{Q}$), which consequently adds more complexity to the calculations. Therefore, it is expected that the exotherm network exhibits a poorer performance as it requires to handle more complexity in learning the feature representations needed for estimating the response.

5.2. *Effect of sequential unfreezing*

Next, the effect of the frozen layers on the performance of the pro-

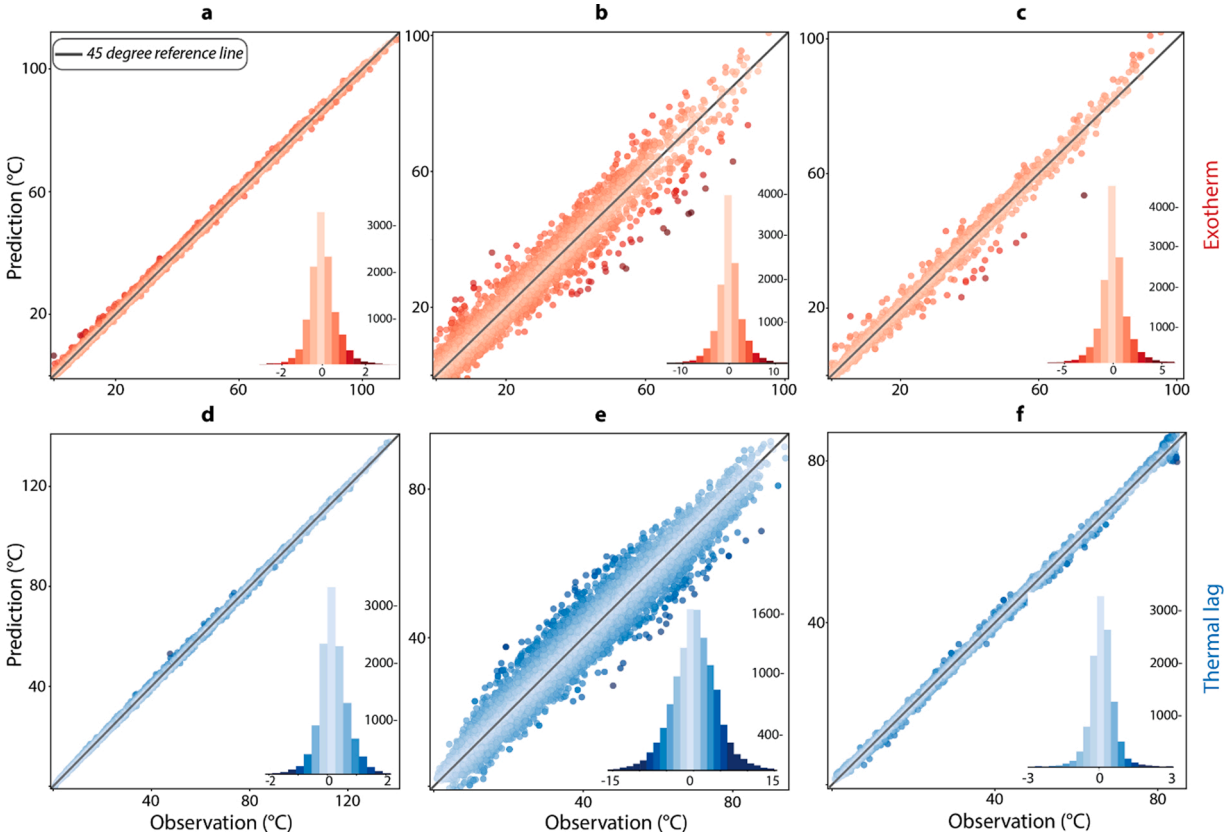

Fig. 5. Pareto plots of exotherm (top row) and thermal lag (bottom row) prediction models on the target CC2 data: (a,d) upper bound (idealistic) models trained on full dataset (for reference only); (b,e) conventional models developed in the presence of limited data; (c,f) TL models in the presense of limied data. Points with darker colors represents a higher deviation between the predicted and true values. Frequency distributions of error values (*prediction – observation*) are also demonstrated.

posed TL method was investigated. After being randomly initialized, the last layer of the network was trained while the rest of the layers were remained unchanged (epochs 1 to n). During the second phase of the sequential unfreezing approach (epochs $n+1$ to $n+m$), a decision needs to be taken on the number of the layers to unfreeze. A non-optimal choice would cause either overfitting or underfitting, depending on the target training sample size and the model complexity. To identify the optimum model setting, the network was trained under different scenarios, each with a specific number of frozen hidden layers (0–5). The generalization performance of these models is compared to that of a network in which sequential unfreezing is not implemented (NN; both transferred and randomly initialized layers are updated from epoch 1 to m). For each model, the training process was carried on until the model converged or when the performance on the validation dataset began to degrade. To achieve this, the early stopping regularization method [50] was implemented. As outlined in Algorithm 1, the method controls the capacity of the network by determining how many epochs the model requires for fitting the training dataset while avoiding overfitting (that is, when exposed to a small set of data, the model, along with the true underlying pattern of the data (signal), captures the residual variations (noise), yielding possibly an unexpected performance on unseen data). In comparison with other common regularization methods, early stopping provides less computational costs as it does not add any additional term to the model cost function [51]. The results were captured and demonstrated in Fig. 6 and Table 3.

Algorithm 1 *- Early stopping (adopted from [43])*

Split the training set (X, Y) into (X_{train}, X_{valid}) and (y_{train}, y_{valid}), respectively.

Train the model using (X_{train}, y_{train}), and update the randomly initialized weights θ for n steps, where n denotes the number of steps between each evaluation process.

Evaluate the model using (X_{valid}, y_{valid}) after training for n steps. If a lower validation error, $ValidationError(\theta)$, is achieved, update the weights. Otherwise, if the model does not improve for p(denoting "patience" parameter) consecutive evaluations, end the training process.

Output the model with the optimal weights $\boldsymbol{\theta}^*$ yielding the lowest validation error

The first observation from Table 3 is that as the number of frozen layers grows, the convergence of the model takes longer. In other words, as the number of trainable parameters decreases, the model becomes simpler and thus requires more iterations to embed the underlying characteristics of the data in the fewer degrees of freedom (neurons). In addition, it is apparent that by increasing the number of frozen layers, the flexibility of the model for adapting to the new task declines, which in turn increases the training and validation losses. Among the evaluated models, the networks with 0 and 1 frozen layers (0FL and 1FL) outputted the lowest training losses. Such behavior is expected, as the act of unfreezing more hidden layers increases the capacity of the transferred model. Therefore, in this study, the 0FL structure is used for evaluating the performance of the proposed TL framework.

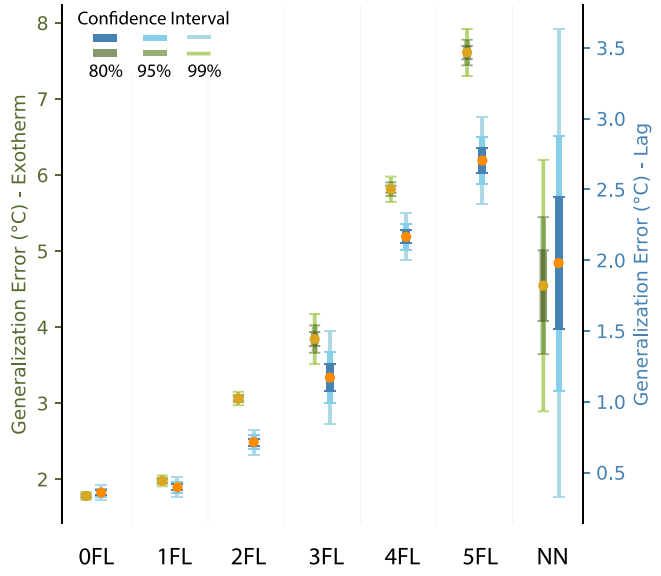

Fig. 6. Comparison of the exotherm (green) and lag (blue) models performance using different settings of sequential unfreezing (zero (0 F L) to five (5 F L) frozen layers) and regular neural network training (NN). Repeatability of the models performance is demonstrated by confidence interval bars (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

Table 3
Result summary of training process for different settings of sequential unfreezing approach. RMSE (MSE) values are reported for the training and validation losses.

Model	Training Loss (°C)		Validation Loss (°C)		Training time (s)	
	Exotherm	Lag	Exotherm	Lag	Exotherm	Lag
0FL	0.182 (0.033)	0.184 (0.034)	**1.603 (2.57)**	0.289 (0.084)	**53**	**98**
1FL	**0.163 (0.027)**	**0.117 (0.014)**	1.893 (3.583)	**0.211 (0.045)**	63	117
2FL	0.626 (0.392)	0.120 (0.014)	3.524 (12.419)	0.229 (0.052)	55	110
3FL	0.854 (0.792)	0.229 (0.052)	3.121 (9.741)	0.488 (0.238)	69	140
4FL	1.903 (3.621)	0.922 (0.85)	6.006 (36.072)	1.533 (2.350)	85	177
5FL	5.192 (26.957)	1.291 (1.667)	10.71 (114.704)	2.175 (4.731)	94	252

Fig. 6 represents the generalizability of the examined networks against unseen (test) data. Both exotherm and lag predictive models exhibit the highest levels of accuracy when fewer layers are kept frozen in the second step of the sequential unfreezing. Besides, the effect of sequential unfreezing on the repeatability of the models' generalization performance is demonstrated by visualizing the confidence intervals. Generally speaking, models armed with sequential unfreezing experience a more stable performance response as opposed to networks with a regular training procedure (NN) in which a high variability in the generalization loss values is evident. This can be due to the effect of a large error gradient generated by the randomly initialized layer which destructively modifies the weights of the transferred layers (catastrophic forgetting) and causes the network to behave unstably. It is shown that sequential unfreezing was able to successfully hinder the effect of the error gradients and stabilize the model's training process. Besides, models with a higher number of frozen levels (i.e., 4 F L and 5 F L) tend to exhibit worse generalizability in comparison with regular training approaches (NN). This suggests that reducing the model's capacity (by fixing the majority of the weights in the network) has a more destructive effect on the ability of the model to properly fit the data than the large error gradients resulting from the random weights of the regressor layer. Finally, the observed large performance variation in the models with more frozen layers (e.g., 3FL, 4FL, and 5FL) is believed to be caused by the inability of the models to converge to a global optimum due to the limited number of trainable layers. This magnifies the randomness in the model behavior and thus would result in a higher variability in the generalization error.

5.2.1. *Model evaluation and trade-off analysis*

The trained TL (sub)models underwent a two-stage performance evaluation process: (i) visual inspection of the model generalization performance via plotting the Pareto plots and absolute error, and (ii) calculating the performance measures, Root Mean Square Error (RMSE) and R-squared (R^2), to rank the models with respect to the quantitative measurements (as described in Section 4.1). In addition, the framework outputs a design chart that takes into account the key factors in manufacturing decision making tasks by visually depicting a trade-off between the model performance, training time, available training

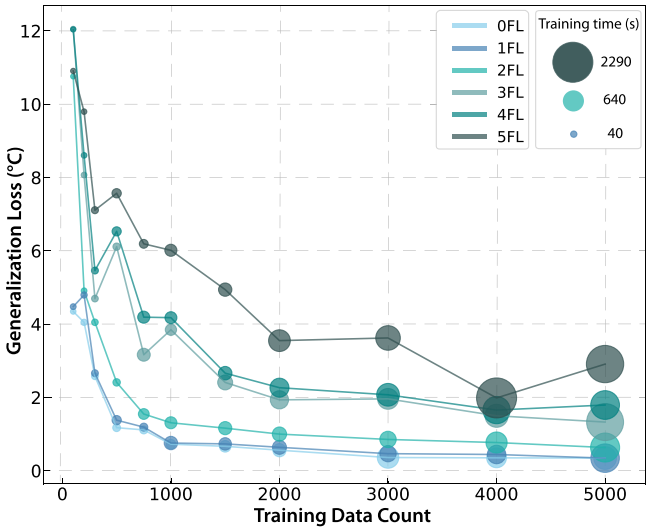

Fig. 7. A general design (trade-off) chart for identifying proper sample size in the proposed composites manufacturing TL framework, against the different number of frozen layers during the transfer. Bubbles'size corresponds to training time. Y-axis represents the models' generalization loss.

data, and model structure. Fig. 7 demonstrates a design chart that may be used to find the optimal network architecture (number of frozen layers) and the number of data points required from a new cure cycle in order to develop (adapt) a reliable predictive tool. The size of the points demonstrates the magnitude of the training time, a vital criterion in time-sensitive decision-making tasks. Generally speaking, the generalization loss declines as the training dataset becomes larger, while models with fewer frozen layers achieve higher performance in shorter periods.

6. Conclusion

Reliable expertise and understanding pertaining to the characteristics of the given material system and the process parameters are traditionally known to be of paramount importance when it comes to designing and implanting advanced composite manufacturing. Due to the high levels of complexity and severe uncertainty (e.g., from the fiber architecture variation to changes in the matrix content and formulation, to uncertainties in controlled and uncontrolled process variables [52]), however, such resources might sometimes fall short in critical decision-making scenarios (e.g., when developing a new, non-classic cure cycle). Therefore, industrial manufacturers, especially for high-cost production cases, tend to avoid the high risk-reward ratio strategies and stick to more conservative designs that lack the crucial element of leveraging the maximum structural and environmental capacity of composite materials [53]. One way to resolve this shortcoming, is the use of data-driven predictive (ML) models for extracting useful insights from historical data for making informed decisions for new designs. This can significantly reduce the design costs by e.g., determining the optimal set of parameters for the new process and avoiding the potential risks (e.g. defects) upfront [20]. However, a drawback with conventional ML approaches is that if the new (targeted) process has major differences in terms of its features (e.g. having a two-hold cure cycle compared to the traditional one-hold cycle), it fails rapidly and cannot be generalized for practical use. On the other hand, in practice, there would be very limited data (trial) from the targeted new process.

In this article, a TL framework for advanced smart manufacturing processes was proposed and exemplified for autoclave composite manufacturing, where the historically abundant data from a past process was transferred (combined with) the limited data from the new process and train an accurate ML model for the new process. In particular, it was demonstrated that utilizing the sequential unfreezing method in the above TL can minimize the chance of model overfitting to limited and small datasets while maintaining the learned knowledge from the source task and avoiding catastrophic forgetting. Analysis of the results showed a major decrease of 88% in the generalization loss using the proposed model in the autoclave processing example. It was shown that there exists a correlation between the model performance on unseen data and the number of frozen layers during the training process. The optimal number of frozen layers can be determined by taking into account the available data for training as well as the complexity of the model. Fewer frozen layers were more effective in maximizing the effect of sequential unfreezing while models with more frozen layers showed less adaptability to new tasks.

Future analysis may be carried out in order to exploit the optimal specifications for the manufacturing process, based on the trained prediction model. Concretely, the developed TL framework can be implemented to create a design space (i.e., by generating reliable predictions) and then a Multiple-Criteria Decision Analysis (MCDA) can be utilized to explore the generated space, seeking Pareto process parameters [54]. The effect of the network architecture on the performance of the lag and exotherm models could also be investigated. Additionally, studying the impact of the number of frozen layers on the model performance variation can be another potential future work direction. Finally, it is believed that the performance of any TL model is dependent on the adequacy of the transferred knowledge from the source task. Limited or unbalanced source data can convey faulty knowledge and therefore, results in poor performance in the target model (negative transfer [37]). Measuring the similarity of the source and target distributions (i.e., using Maximum Mean Discrepancy or KL-divergence [51]) before transferring the knowledge can be considered as a highly viable solution for identifying suitable source knowledge.

Declaration of Competing Interest

The authors report no declarations of interest.

Acknowledgments

The authors would like to thank the support and helpful comments from the colleagues at the Composites Research Network (CRN), University of British Columbia and German Aerospace Center (DLR). This study was financially supported by the New Frontiers in Research Fund – Exploration stream (award number: NFRFE-2019-01440).

References

[1] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. https://doi.org/10.1038/nature14539.
[2] Kusiak A. Smart manufacturing. Int J Prod Res 2018;56:508–17. https://doi.org/10.1080/00207543.2017.1351644.
[3] Kusiak A. Smart manufacturing must embrace big data. Nature 2017;544:23–5. https://doi.org/10.1038/544023a.
[4] Subramaniyan M, Skoogh A, Muhammad AS, Bokrantz J, Johansson B, Roser C. A generic hierarchical clustering approach for detecting bottlenecks in manufacturing. Int J Ind Manuf Syst Eng 2020;55:143–58. https://doi.org/10.1016/j.jmsy.2020.02.011.
[5] Yang H, Kumara S, Bukkapatnam STS, Tsung F. The internet of things for smart manufacturing: a review. IISE Trans 2019;51:1190–216. https://doi.org/10.1080/24725854.2018.1555383.
[6] Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. J Manuf Syst 2018;48:157–69. https://doi.org/10.1016/j.jmsy.2018.01.006.
[7] Lu Y, Xu X, Wang L. Smart manufacturing process and system automation – a critical review of the standards and envisioned scenarios. J Manuf Syst 2020;56:312–25. https://doi.org/10.1016/j.jmsy.2020.06.010.
[8] Fallahzadeh R, Ghasemzadeh H. Personalization without user interruption. In: Proc. 8th Int. Conf. Cyber-Physical Syst.; 2017. p. 293–302. https://doi.org/10.1145/3055004.3055015.
[9] Guo L, Lei Y, Xing S, Yan T, Li N. Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabeled data. IEEE Trans Ind Electron 2019;66:7316–25. https://doi.org/10.1109/TIE.2018.2877090.
[10] Arian Nik M, Fayazbakhsh K, Pasini D, Lessard L. A comparative study of metamodeling methods for the design optimization of variable stiffness

composites. Compos Struct 2014;107:494–501. https://doi.org/10.1016/j.compstruct.2013.08.023.
[11] Zimmerling C, Dörr D, Henning F, Kärger L. A machine learning assisted approach for textile formability assessment and design improvement of composite components. Compos Part A Appl Sci Manuf 2019;124:105459. https://doi.org/10.1016/j.compositesa.2019.05.027.
[12] Balokas G, Czichon S, Rolfes R. Neural network assisted multiscale analysis for the elastic properties prediction of 3D braided composites under uncertainty. Compos Struct 2018;183:550–62. https://doi.org/10.1016/j.compstruct.2017.06.037.
[13] Chen C-T, Gu GX. Machine learning for composite materials. MRS Commun 2019;9:556–66. https://doi.org/10.1557/mrc.2019.32.
[14] Khan A, Ko DK, Lim SC, Kim HS. Structural vibration-based classification and prediction of delamination in smart composite laminates using deep learning neural network. Compos Part B Eng 2019;161:586–94. https://doi.org/10.1016/j.compositesb.2018.12.118.
[15] Yang Z, Yabansu YC, Al-Bahrani R, Liao W, Choudhary AN, Kalidindi SR, et al. Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. Comput Mater Sci 2018;151:278–87. https://doi.org/10.1016/j.commatsci.2018.05.014.
[16] Rong Q, Wei H, Huang X, Bao H. Predicting the effective thermal conductivity of composites from cross sections images using deep learning methods. Compos Sci Technol 2019;184:107861. https://doi.org/10.1016/j.compscitech.2019.107861.
[17] Pan SJ, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng 2010;22:1345–59. https://doi.org/10.1109/TKDE.2009.191.
[18] Huang S, Guo Y, Liu D, Zha S, Fang W. A two-stage transfer learning-based deep learning approach for production progress prediction in IoT-Enabled manufacturing. IEEE Internet Things J 2019;6:10627–38. https://doi.org/10.1109/JIOT.2019.2940131.
[19] Ren H, Liu W, Shan M, Wang X. A new wind turbine health condition monitoring method based on VMD-MPE and feature-based transfer learning. Measurement 2019;148:106906. https://doi.org/10.1016/j.measurement.2019.106906.
[20] Fabris J, Lussier D, Zobeiry N, Mobuchon C, Poursartip A. In: Development of Standardized Approaches to Thermal Management in Composites Manufacturing. SAMPE Tech. Conf. Proceedings. Soc. Adv. Mater. Process Eng.; 2014. p. 2–5.
[21] Sozer EM, Advani SG. Process modeling in composites manufacturing. 2nd ed. Boca Raton, Florida: CRC press; 2010.
[22] Astrom BT. Manufacturing of polymer composites. 1st ed. London: Routledge; 2018.
[23] Fernlund G, Mobuchon C, Zobeiry N. 2.3 Autoclave processing. Compr Compos Mater II 2018;2:42–62. https://doi.org/10.1016/b978-0-12-803581-8.09899-4.
[24] Campbell Jr FC. Manufacturing processes for advanced composites. 1st ed. Oxford: Elsevier; 2003.
[25] Padmanabhan SK, Pitchumani R. Stochastic analysis of isothermal cure of resin systems. Polym Compos 1999;20:72–85. https://doi.org/10.1002/pc.10336.
[26] Potter KD. Understanding the origins of defects and variability in composites manufacture. In: Int. Conf. Compos. Mater. (ICCM)-17; 2009. p. 18–25.
[27] Khayyam H, Fakhrhoseini SM, Church JS, Milani AS, Bab-Hadiashar A, Jazar RN, et al. Predictive modelling and optimization of carbon fiber mechanical properties through high temperature furnace. Appl Therm Eng 2017;125:1539–54. https://doi.org/10.1016/j.applthermaleng.2017.06.071.
[28] Wuest T, Weimer D, Irgens C, Thoben K-D. Machine learning in manufacturing: advantages, challenges, and applications. Prod Manuf Res 2016;4:23–45. https://doi.org/10.1080/21693277.2016.1192517.
[29] Wang J, Ma Y, Zhang L, Gao RX, Wu D. Deep learning for smart manufacturing: methods and applications. Int J Ind Manuf Syst Eng 2018;48:144–56. https://doi.org/10.1016/j.jmsy.2018.01.003.
[30] Chen M, Mao S, Liu Y. Big data: a survey. Mob Networks Appl 2014;19:171–209. https://doi.org/10.1007/s11036-013-0489-0.
[31] Yang B, Lei Y, Jia F, Xing S. An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings. Mech Syst Signal Process 2019;122:692–706. https://doi.org/10.1016/j.ymssp.2018.12.051.
[32] Zhang K, Chen J, Zhang T, He S, Pan T, Zhou Z. Intelligent fault diagnosis of mechanical equipment under varying working condition via iterative matching network augmented with selective Signal reuse strategy. J Manuf Syst 2020;57:400–15. https://doi.org/10.1016/j.jmsy.2020.10.007.
[33] Xu Y, Sun Y, Liu X, Zheng Y. A digital-twin-assisted fault diagnosis using deep transfer learning. IEEE Access 2019;7:19990–9. https://doi.org/10.1109/ACCESS.2018.2890566.
[34] Sun C, Ma M, Zhao Z, Tian S, Yan R, Chen X. Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing. IEEE Trans Ind Informatics 2019;15:2416–25. https://doi.org/10.1109/TII.2018.2881543.
[35] Ferguson M, Ak R, Lee Y-TT, Law KH. Detection and segmentation of manufacturing defects with convolutional neural networks and transfer learning. Smart Sustain Manuf Syst 2018;2:20180033. https://doi.org/10.1520/ssms20180033.
[36] Li X, Zhang W, Ma H, Luo Z, Li X. Deep learning-based adversarial multi-classifier optimization for cross-domain machinery fault diagnostics. J Manuf Syst 2020;55:334–47. https://doi.org/10.1016/j.jmsy.2020.04.017.
[37] Pan SJ, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng 2010;22:1345–59. https://doi.org/10.1109/TKDE.2009.191.
[38] Weiss K, Khoshgoftaar TM, Wang D. A survey of transfer learning. J Big Data 2016;3:9. https://doi.org/10.1186/s40537-016-0043-6.
[39] Crawford B, Khayyam H, Milani AS, Jazar RN. Big data modeling approaches for engineering applications. Nonlinear Approaches Eng. Appl.. 1st ed. Cham: Springer; 2020. p. 307–65.
[40] RAVEN simulation software 2016. http://www.convergent.ca/products/raven-simulation-software.
[41] Ciriscioli PR, Springer GS, Lee Il W. An expert system for autoclave curing of composites. J Compos Mater 1991;25:1542–87. https://doi.org/10.1177/002199839102501201.
[42] Neyshabur B, Sedghi H, Zhang C. What is being transferred in transfer learning? ArXiv Prepr ArXiv200811687 2020.
[43] Bastani K, Asgari E, Namavari H. Wide and deep learning for peer-to-peer lending. Expert Syst Appl 2019;134:209–24. https://doi.org/10.1016/j.eswa.2019.05.042.
[44] Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks. Int Conf Mach Learn 2017:1126–35.
[45] French R. Catastrophic forgetting in connectionist networks. Trends Cogn Sci 1999;3:128–35. https://doi.org/10.1016/S1364-6613(99)01294-2.
[46] Chronopoulou A, Baziotis C, Potamianos A. An embarrassingly simple approach for transfer learning from pretrained language models. In: Proc. 2019 Conf.; 2019. p. 2089–95. https://doi.org/10.18653/v1/N19-1213.
[47] Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? Adv Neural Inf Process Syst 2014;4:3320–8.
[48] Klambauer G, Unterthiner T, Mayr A, Hochreiter S. Self-normalizing neural networks. Adv. Neural Inf. Process. Syst.. 2017. p. 971–80.
[49] Johnston AA. An integrated model of the development of process-induced deformation in autoclave processing of composite structures. University of British Columbia; 1997. https://doi.org/10.14288/1.0088805.
[50] Prechelt L. Early stopping-but when? Neural Networks: tricks of the trade. 1st ed. Berlin, Heidelberg: Springer; 1998. p. 55–69.
[51] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT press; 2016.
[52] Endruweit A, Long AC. Influence of stochastic variations in the fibre spacing on the permeability of bi-directional textile fabrics. Compos Part A Appl Sci Manuf 2006;37:679–94. https://doi.org/10.1016/j.compositesa.2005.08.003.
[53] Mesogitis TS, Skordos AA, Long AC. Uncertainty in the manufacturing of fibrous thermosetting composites: a review. Compos Part A Appl Sci Manuf 2014;57:67–75. https://doi.org/10.1016/j.compositesa.2013.11.004.
[54] Ramezankhani M, Crawford B, Khayyam H, Naebe M, Seethaler R, Milani AS. A multi-objective Gaussian process approach for optimization and prediction of carbonization process in carbon fiber production under uncertainty. Adv Compos Hybrid Mater 2019;2:444–55. https://doi.org/10.1007/s42114-019-00107-6.